represent (M by M) positive definite matrixes, C an (M by N) matrix, D an (N by N) positive definite matrix, an inverse matrix of A is given by

$$A^{-1}=B-BC(D+C^H BBC)^{-1}C^H B \qquad (33)$$

Applying this Lemma to the calculation of the inverse matrix in Eq. (32),

5       $H_L(n)H_L^H(n)+\sigma^2 I=B^{-1}+CD^{-1}C^H$,

$H_L(n)H_L^H(n)=CD^{-1}C^H$,

$\sigma^2 I=B^{-1}$,

$h(n)=C$,

$D^{-1}=I$,

10       $H_L^H(n)=C^H$

By using them to calculate Eq. (33), the inverse matrix calculation in Eq. (32) is solved. Incidentally, Eq. (33) also contains an inverse matrix calculation $(D+C^H BBC)^{-1}$, but this inverse matrix is also scalar, and hence it can similarly be calculated.

15   The thus calculated tap coefficient value G(n) is used to perform linear filtering of the difference value $R_c(n)$ by the transversal filter 111, and the filtering result Z(n) and the estimated channel impulse response value $H_L(n)$ stored in the storage part 106B are used to calculate Eq. (25) in the soft decision value calculating part 120 to derive the soft decision value $\Lambda_1[b(n)]$.

20   The equivalent amplitude $\mu(n)$ in Eq. (25) is given by the following equation by substituting Eq. (31) for $\Lambda(n)$ in Fig. (23).

$$\mu(n) = [H_L(n)^H[H_L(n)H_L(n)^H +\sigma^2 I]^{-1}H_L(n)]$$
$$=H_L(n)^H G(n) \qquad (34)$$

Incidentally, in the first iteration of equalization the tap coefficient calculating

25   part 112 of the linear filter part 110 may also use the Matrix Inversion Lemma to calculate the tap coefficient G(n).

As described above, this embodiment permits reduction of the

number M needs to be at least 2 or more.

Assuming that the filter output can be approximated by a Gaussian distribution, Z(n) by Eq. (21) can be described as follows (see V. Poor and S. Verdu, "Probability of Error in MMSE Multiuser Detection," IEEE Trans. Information Theory., vol. IT-43, No. 3, pp. 858-871, May 1997).

$$Z(n) = \mu(n)b(n) + \eta(n) \tag{22}$$

where $\mu(n)$ is an equivalent amplitude of the output signal and $\eta(n)$ represents a Gaussian distribution of an average 0 and the variance $v^2(n)$. Therefore, $\mu(n)$ and $v(n)$ can be expressed as follows:

$$\mu(n) = E\{Z(n)b(n)\}$$

$$= [H_m(n)^H[H_m(n)\Lambda(n)H_m(n)^H + \sigma^2 I]^{-1}H_m(n)]_{J,J} \tag{23}$$

where $[]_{J,J}$ indicates the element at the intersection of the J-th row and the J-th column of the matrix.

$$v^2 = \text{var}\{Z(n)\} = \mu(n) - \mu^2(n) \tag{24}$$

From the above, the extrinsic information derived by the linear equalizer can be deduced from the following equation.

$$\lambda_1[b(n)] = \log\frac{p(Z(n)|b(n) = +1)}{p(Z(n)|b(n) = -1)} = \frac{4\text{Real}\{Z(n)\}}{1 - \mu(n)} \tag{25}$$

To obtain the optimum tap coefficient G(n) with the above-described method, however, it is necessary that the inverse matrix calculation expressed by the following equation be conducted in Eq. (20) for each point in time; hence, much time is required for calculation.

$$\Phi(n) = [H_m(n)\Lambda(n)H_m(n)^H + \sigma^2 I]^{-1} \tag{26}$$

With the conventional method for iterative equalization by the linear equalizer, the inverse matrix calculation needs to be conducted for each point in time to update the tap coefficient—this gives rise to the problem of high computational complexity.